Project: State of Washington Multi-Agency Forest Project

Title: **Business Requirements for Windows 2000 Administrators Communication Plan**

Version: 1.0

Status: Draft for Comment

Date: August 31, 2001

State of Washington

Windows 2000 Forest Project

Business Requirements for Windows 2000 Administrators Communication Plan

Document Information

Title: Technical Communication Plan for Connected Administrators

Status: In Development

Authors: Anthony Witecki, Microsoft Consulting Services
John Ditto, Department of Information Services

Reviewers: Forest Resource Group, Steering Committee.

Document History

WHEN	WHO	WHAT
June 27, 2001	Anthony Witecki	Initial Document Development
August 21, 2001	Jane Rasmussen	Review by MCS Managing Consultant
August 30, 2001	Anthony Witecki	Initial document draft. Has been reviewed by Forest Resource Group and Steering Committee.
September 5, 2001	John Ditto	Entered into the Document Approval Process as Draft for Comment.

Table of Contents

Objective

This document defines the procedures by which the following types of information are communicated in the multi-agency distributed Windows 2000 environment. Communication includes, but would not be limited to:

1. Specific incidents and their resolution
2. Common problems and easy fix suggestions (Tips and Tricks)
3. Incident and problem trends
4. Potential service disruption warnings
5. Self service suggestions and instructions
6. Current service status (systems that are down)
7. Training materials
8. Any other relevant information
9. Standard procedures forms and documentation

Purpose of This Document

The purpose of this document is to outline some of the business requirements for a strategic communication plan. This document does use some technical terms and diagrams for purposes of clarification of the business requirements or conceptual design.

The figures used in this document are for clarification purposes only. They do not represent a technical design recommendation. All figures are subject to change during the actual design process.

General Overview

Business Goal

The purpose of a communication plan is to provide a means for system administrators, technical managers, and connected end-users to share information relevant to the State's Windows 2000 environment.

To accomplish this goal, a series of high-level business objectives have been identified to define the information to be communicated, the users, their needs, and the process by which information is authored, aggregated, and disseminated.

Business Challenges

The following challenges have been identified, specific to the State environment acting as a single enterprise under a Windows 2000 Single Root Forest.

1. **Disparate Networks and Technologies.** Beginning in 2000, the State of Washington undertook a major initiative to move to a single enterprise model. The transition period may take several years, during which agencies will be operating on a variety of dissimilar platforms working at various stages of deployment. A single, globally available resource must exist to facilitate communication.
2. **Authority is delegated.** Each of more than 100 agencies maintains their own domain and site infrastructure. This provides LAN administrators with autonomy, but brings with it the risk of duplicated efforts. The best way to mitigate this risk is to improve the communication among agency administrators so that knowledge is learned once and read many times.
3. **Enterprise Models Bring Complexity.** The process of making changes in a large environment requires several phases of planning, approval and implementation in lab, test and production environments. Often, the ownership of issues changes hands quickly, making an audit trail difficult to follow. Large organizations typically report that "finding information" is their employees' number one problem.

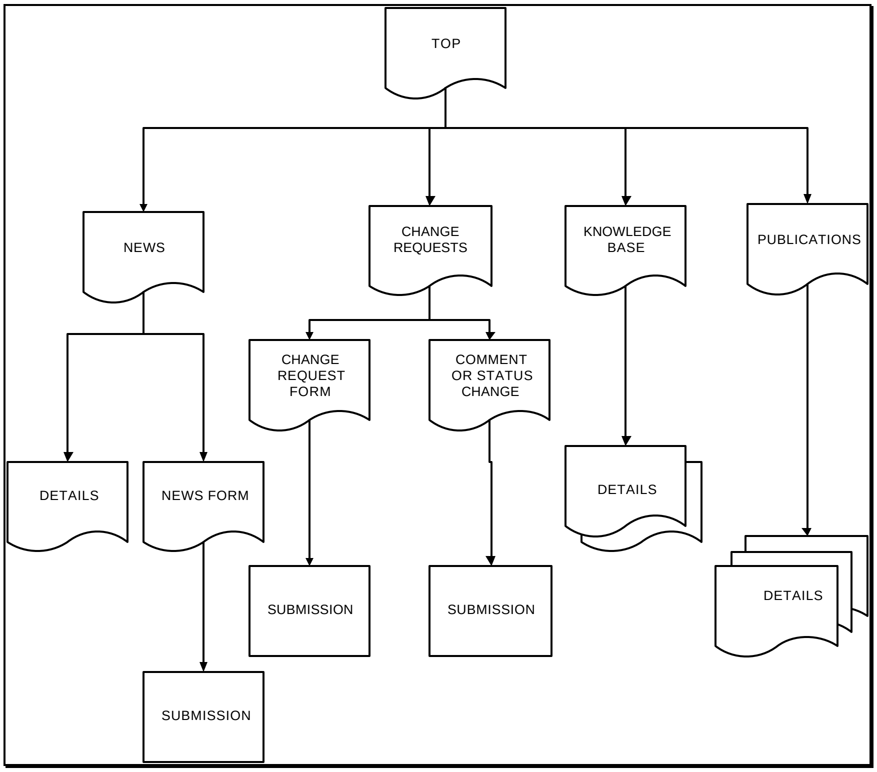


Figure 1 – Information Requirements Overview

Business Objectives

To meet these challenges, the following business objectives for Intra-Forest communications have been defined. Figure 1 depicts these objectives at a high level. Each objective is elaborated in subsequent sections.

1. **Users are grouped into special interest groups, or communities.** These communities are based on the organizational objectives of the Windows 2000 initiative. Table 1 defines the communities and what their specific communication needs are. Users may belong to one or more communities and must be able to subscribe and unsubscribe from communities at will.
   a. Each community has one or more publishers, responsible for managing content in that community.
   b. Publishers have the option of limiting membership in their communities to appropriate personnel.
   c. Publishers have the option of securing content in their communities, limiting it to the community's members.
   d. The Forest Resource Group has considered the possibility that this tool might be used for communication vehicles other than the Windows 2000 infrastructure. Should that decision be made, it would be preferable to add an additional layer of community

nesting to accommodate multiple environments (infrastructure, forest-wide applications, etc.).

2. **Users are classified within each community according to roles.** Roles define the actions that a user can take within a community. A user can belong to one or more roles and belong to different roles across different communities. For example, a user might be a reader in one community, and an author in another. Identified roles and their descriptions are listed in Table 2.
3. **Community members have one location for all information.** Because the information tied to communities is not agency-specific, it is imperative that users of the communities retrieve that information from a single source.
4. **News and communication is organized by community.** The Windows 2000 communication infrastructure is not intended for agency-specific collaboration. Therefore, published news must be globally interesting and must be associated with one or more communities. Figure 2 provides a graphical representation of the relationship among communities (categories), news, and users.
5. **Authoritative documents are separately published.** Because they represent the culmination of work from several people at several levels across several agencies, authoritative documents will be published and stored separately from news and other communications.
6. **Change requests are accessible from the communication tool, but separately maintained and accessed.** Several users of the communications platform will be interested only in viewing the queue of change requests, or obtaining status about their own change requests. For this reason, the change management tool will be handled separately from standard communications.
7. **Technical information will be stored in its own knowledge management repository.** Technical information usually has a longer life span than news and status reports concerning the environment. Technical information includes, but is not limited to, documented PSS incidents, HOWTO documents, and best practices. It will be maintained separately.

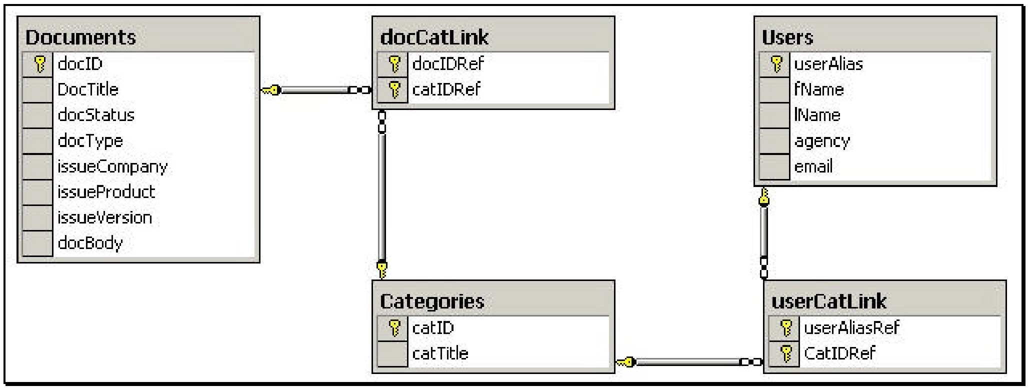


Figure 2 - Relationship among Users, Documents and Categories

Individual or Group	Purpose and Needs
Change Manager Group	The individual(s) that have been appointed or assigned responsibility and oversight of the change management process. They need notification of new change requests and an avenue to communicate those approvals or escalations with the Forest Resource Group. They also need to communicate impact and status to connected forest administrators.
Forest Resource Group	This group is responsible for enterprise-level management of the forest. They need to know about significant or major changes and planning issues. They are also responsible for writing and approving the documentation that will typically be communicated.
Connected Admins Group	This group includes all network administrators that are connected in production. They need access to news about scheduled downtime, proposed changes and associated timelines, environment status, and contacts. They will contribute information in the form of best practices, troubleshooting tips and incident resolution.
Connected AppDevs	Senior Application Developers throughout the connected forest. This group needs information about strategic direction and new service offerings from DIS and the Windows 2000 Forest.
Forest IT Managers	Connected IT managers throughout the Forest. Communication is generally less technical, higher level.
Steering Committee	Similar needs to the forest IT Managers group, but only for those members selected to participate on the Steering committee for the project.
Security Administrators	Connected users and IT staff that are concerned with the security of Windows 2000 network. Need the ability to communicate security issues, post follow-ups, recommend solutions. May contain non-IT folks, such as State Auditor, as members.
Authenticated Users (Forest Wide)	Long-term, this group would expect to see status reports on enterprise services like Exchange email (for example: reasons why Exchange servers are down).
State Employees	Information useful to any state employee. Such as Background documentation. Nothing related to status or operations gets exposed to this group.
Enterprise Administrators	Need access to server status, replication status and other technical issues, in addition to information in all previous groups.

Table 1 - Identified Communities and Their Purpose

ROLE	DESCRIPTION
Subscribers	Push-based consumers of information. When someone subscribes to information, they agree that such information should be pushed out to them (email being the most common way).
Readers	Pull-based consumers. Readers search for specific information on an as-needed basis. Ideally, readers don't have to search for information; the most important information surfaces based on their profile and historic needs.
Authors	Can create new content and edit content owned by them specifically.
Editors	Editors are responsible for reviewing and editing the work of authors and publishing content into the communication system. Editors are associated with authors in a one-to-many relationship. Editors control content by author, not by category.
Publishers	Publishers are similar to editors except that they control content by category. For that reason, they are excluded from most of the work flow in content publication, but can, at any time, pull issues from their community. Publishers act as a safety net for preventing misinformation in their area of expertise.

Table 2 – Identified Roles

Publishing News

Business Requirements

The following requirements have been identified for the News Publishing Component of the communications platform. News represents the single largest reason a user would use the platform and will likely have the highest level of activity.

1. Centralized Windows 2000 IT administrators must be able to communicate status, scheduled downtime, or changes about to be implemented in the production environment.
2. News related to overall network security (patches, viruses, etc.) can be rapidly distributed to the connected community.
3. Users are identified by the platform, and news is returned to them based on their community membership. Anonymous connections are presented with general information or information contained in anonymous communities.
4. Authors are required to assign new content to one or more communities, ultimately determining which communities will be exposed to the information.
    a. Authors can only assign content to communities that they have an author role or higher.
5. Upon submission, all news must be approved by the author's editor before it can be published.
    a. Editors are considered the "managers" of authors. They can create items in their community, and approve items in their community.
    b. Editors cannot approve items for other communities.
6. Publishers of categories can remove the link between a piece of content and their community at their discretion. Publishers may also link content from other communities to their own.
    a. Publishers can create and edit content in their own community.
    b. Publishers cannot create or edit content from other communities.
7. If an author submits content assigned to more than one community, the editor of each community must approve the association.
8. Roles are used to determine which actions a user can take on the platform, NOT which information is displayed.
9. The information that is communicated must be prioritized to take advantage of screen real-estate. Information should be prioritized as follows:
    a. News is filtered before any sorts to include only the news for a valid date range (which is configurable).
    b. All news flagged as Critical is given priority over High, Normal, Low, respectively.
    c. News is sorted by date descending, making the newest news appear higher than older news.

d. Users must have the means to view communication history, such as catching up after returning from vacation.

10. The creator of a community will decide on the security level for that community. The security of a community takes one of two possibilities:
    a. Restricted access: the creator determines which individuals are assigned to which roles. Any requests by a person to join a role must be approved by the creator of the community.
    b. Open Enrollment: anyone can freely add themselves to the reader role, but additional authors, publishers and editors must be approved by the community creator.

Process Workflow

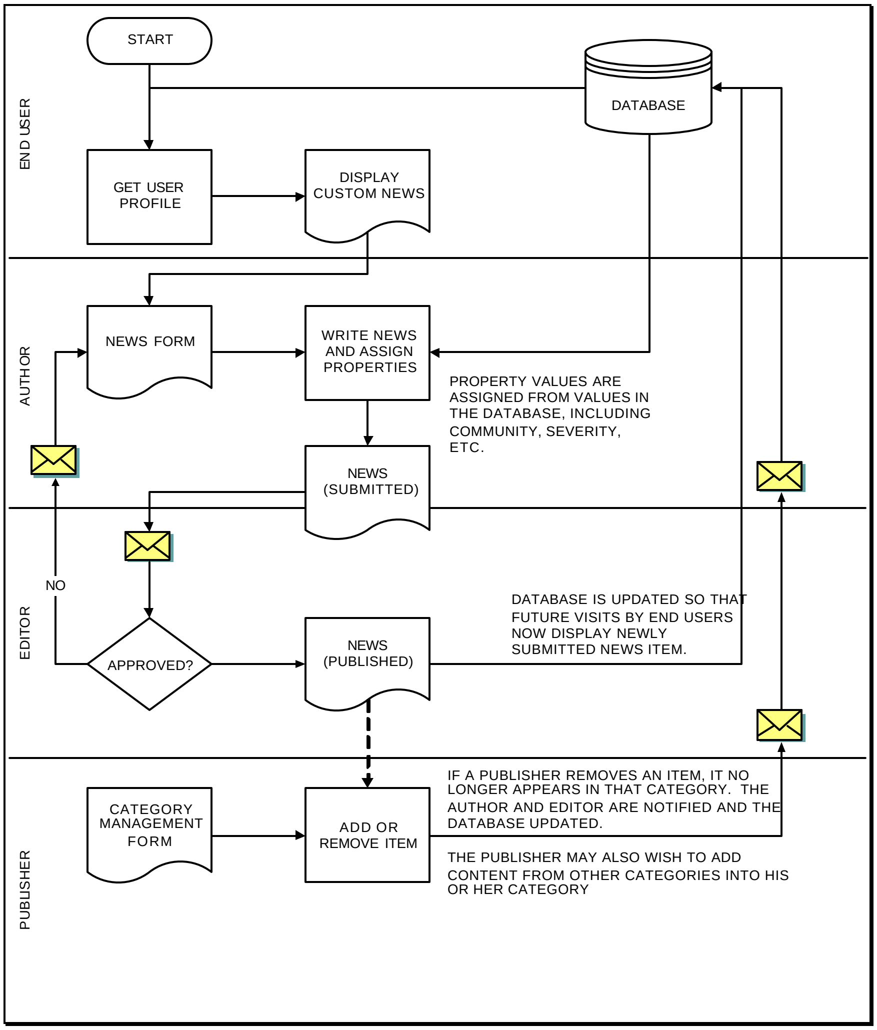

START
DATABASE
END USER
GET USER PROFILE
DISPLAY CUSTOM NEWS
AUTHOR
NEWS FORM
WRITE NEWS AND ASSIGN PROPERTIES
PROPERTY VALUES ARE ASSIGNED FROM VALUES IN THE DATABASE, INCLUDING COMMUNITY, SEVERITY, ETC.
NEWS (SUBMITTED)
EDITOR
NO
APPROVED?
NEWS (PUBLISHED)
DATABASE IS UPDATED SO THAT FUTURE VISITS BY END USERS NOW DISPLAY NEWLY SUBMITTED NEWS ITEM.
PUBLISHER
CATEGORY MANAGEMENT FORM
ADD OR REMOVE ITEM
IF A PUBLISHER REMOVES AN ITEM, IT NO LONGER APPEARS IN THAT CATEGORY. THE AUTHOR AND EDITOR ARE NOTIFIED AND THE DATABASE UPDATED.
THE PUBLISHER MAY ALSO WISH TO ADD CONTENT FROM OTHER CATEGORIES INTO HIS OR HER CATEGORY

Figure 3 – News Publication Workflow

The process by which a **communication item (CI)** flows through the platform is depicted in Figure 3.

1. The user logs onto the platform and is presented with the relevant news based on priority, community-membership and date.
2. If the user is identified as an author, editor, or publisher role, they are given the option to switch into author mode to make edits[1] or submit new content.

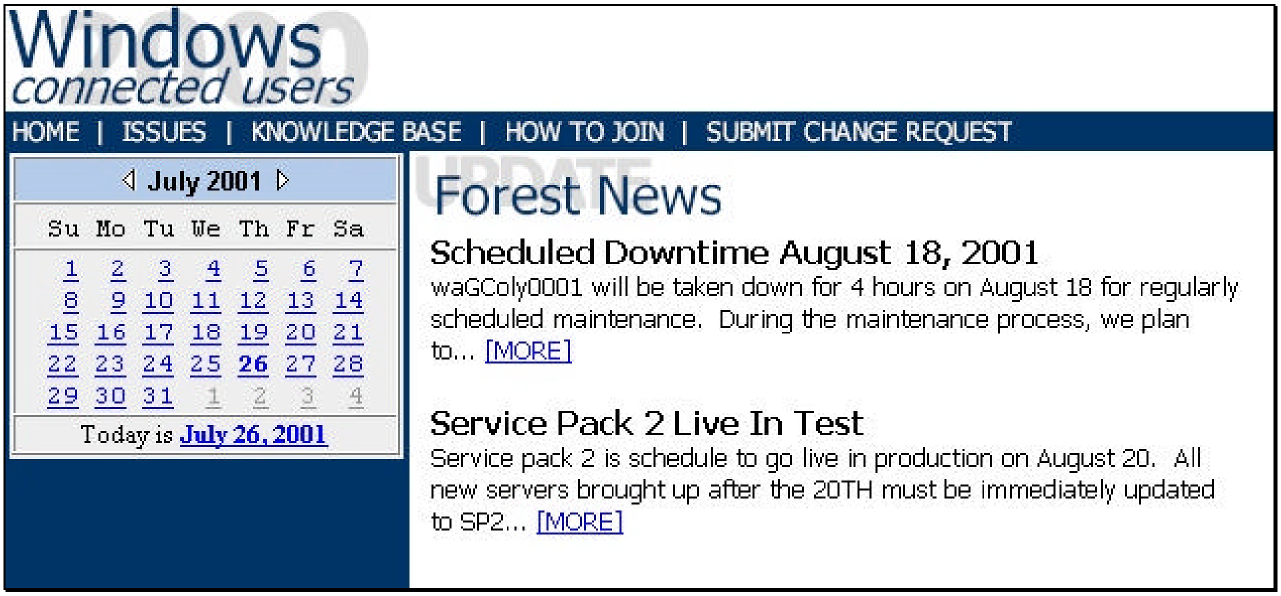
[1] The publisher can create new items and edit any item she submits. However, she cannot make edits to the content of authors outside her community.

3. The author must assign her CI to at least one community. She can only assign content to communities where her role is author, editor, or publisher.
4. Upon submission of the content, the CI is assigned a status of "submitted" and the editor assigned to that author is notified.
5. If the editor approves the content, it is published and the status of the CI is changed to "Published". If the editor denies the submission, the status is changed to "Denied" or "Returned for Edit" and the author is notified.
6. If the editor changes the content and subsequently approves the content, a notification goes to the author showing the original text and the changed text. This ensures that the authors intent is not changed by the editor.
7. At any time, a publisher can remove the association between the CI and his community. Upon doing so, both the author and editor are notified.
8. End Users have access to information as soon as its status is "Published."
9. If a piece of information is pulled from Publish status for edits, its status is changed back to submitted. This ensures that changes are not automatically re-approved until explicitly approved by the editors in each community.

End User Design Considerations

The user experience is critical toward making a communications platform successful. Information should be organized in a manner that is pleasing to the eye, centers the user's attention on both content and tasks simultaneously, and leads the user to participate and return for subsequent visits.

Figure 4 illustrates a possible end-user experience using a web site model. Tasks are organized across the top, the historical data is immediately accessible from the left column and the relevant news stands out and leads the customer to click-thru to down-level detail.


Windows
connected users
HOME | ISSUES | KNOWLEDGE BASE | HOW TO JOIN | SUBMIT CHANGE REQUEST
◁ July 2001 ▷

Su	Mo	Tu	We	Th	Fr	Sa
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30	31	1	2	3	4

Today is July 26, 2001

Forest News

Scheduled Downtime August 18, 2001

waGColy0001 will be taken down for 4 hours on August 18 for regularly scheduled maintenance. During the maintenance process, we plan to... [MORE]

Service Pack 2 Live In Test

Service pack 2 is schedule to go live in production on August 20. All new servers brought up after the 20TH must be immediately updated to SP2... [MORE]

Figure 4 - Sample Communications Platform User Interface

Change Request Monitoring

Business Objectives

DIS currently uses an internal tool, InfoMan, to manage change request. However, this tool does not provide visibility to the process, status, or communications surrounding a particular change outside of the immediate change management group. The business objectives for implementing change management in the communication platform are not necessarily to re-create a system, but rather to provide a means to expose this system to change initiators and interested parties.

The business objectives for the change management monitoring component are as follows:

1. Any change management solution must comply with the policies and procedures defined in the Windows 2000 Change Management Document, approved by the Steering Committee.
2. Users of the change management tool are grouped into two categories:
   a. General Users. General users can submit new **change requests (CR)** and view the status of all change requests. They cannot make edits to existing change requests, except for those they submitted. They must be network analysts or greater in title.
   b. Change Management Team. This includes the change manager, change owners, and approving bodies. They use the tool primarily to update the status and record implementation details that can be ultimately read by the general users and themselves.

3. Change requests should be submitted using a standard form to ensure completeness of required information. All required fields must be completed before the end user can submit the form.
4. The change management user interface should allow readers to view the status of multiple change requests simultaneously.
5. Comments and discussion threads internal to the change management group should be collected and stored with the change request, exposed via the communications platform interface.
6. The change management tool should identify users and automate routine data such as user names, dates, times, and status.
7. Notifications will occasionally be necessary, based on the events identified in Table 3.
8. Approving bodies must be able to approve CRs from within the tool environment.
9. Changes to various fields in the CR should be audited. The following events should be audited:
    a. Changes to the Owner (all owners are retained historically and notified when a request is past due or completed).
    b. Changes to Status.
    c. Changes to User Supplied Information (only options are priority, impact, schema and urgency).
    d. Individuals that have clicked the approve or deny buttons.

Notification Requirements	
Upon Submission of an Request	Sent to Change Manager Acknowledgement sent to initiator
Upon Assignment of a Request	Sent to Change Owner
Upon changing the status to Pending Approval	Sent to members of the approving body (currently the Forest Resource Group or Steering Committee).
Upon Approving For Test and Approval for Production	Sent to connected administrators.

Table 3 - Events Requiring Notification

Process Workflow

Figure 5 illustrates the desired workflow within the change management tool.

1. New **change requests (CR)** are submitted by the end user. The initial status of the CR is "Submitted."

2. As part of his routine, the change manager reviews submissions. Depending on the severity of the CR, he will approve automatically (for simple changes) or escalate for approval. The change manager may wish to post comments prior to escalation.
3. After reviewing the CR, the CR's status is changed to "Reviewed," indicating to the initiator (upon subsequent viewing) that her request has been reviewed, but not approved.
4. Members of the forest resource group can view the details of the change request and individually post follow-up comments and submit their approval from within the tool. This eliminates the need for meetings, but does not preclude them. In some cases, a face-to-face meeting will still take place, but the tool would ultimately tally the votes.
5. Once approved by the Forest Resource Group, the status will be changed to "Pending Approval" if the steering committee must sign off. Otherwise, it will change to "Approved For Test," communicating to network administrators that the changes will be deployed in the test environment.
6. If the steering committee must approve the document, and they do, the CR's status will change to "Approved for Test."
7. After a predetermined time in the test environment (typically 30 days), the change owner will update the status of the CR to "Approved for Production." At this point the CR will be considered closed.

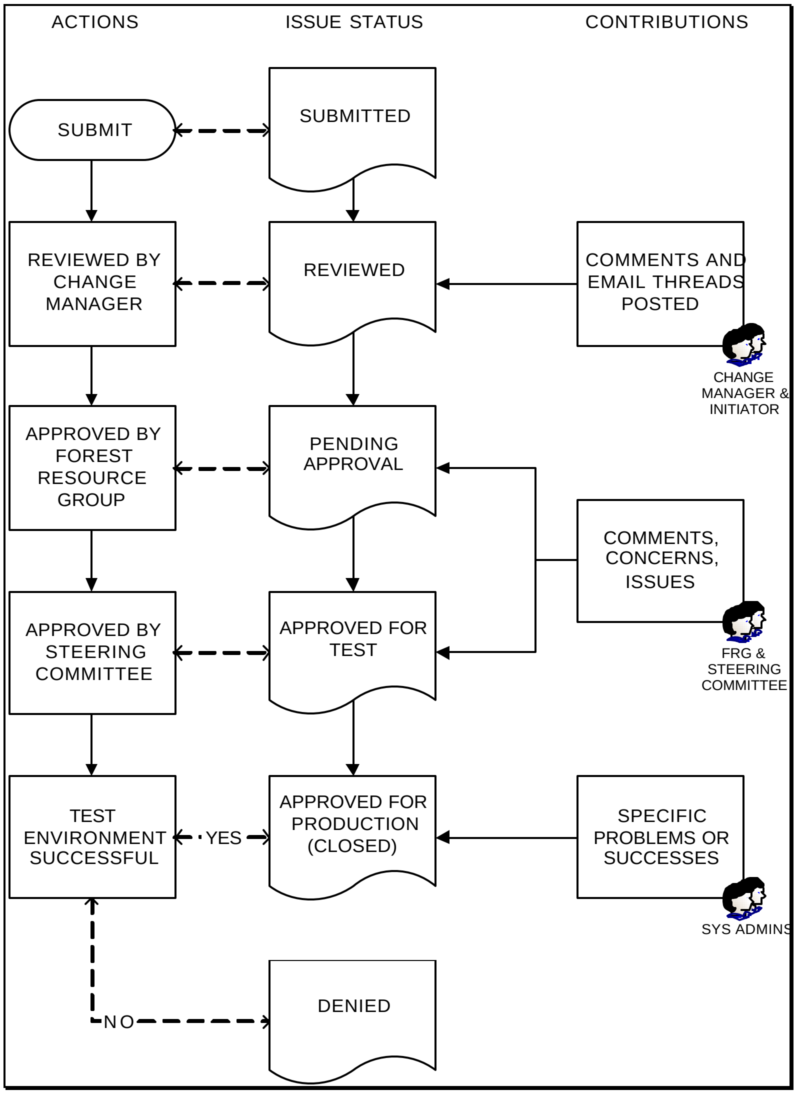

ACTIONS
ISSUE STATUS
CONTRIBUTIONS
SUBMIT
SUBMITTED
REVIEWED BY CHANGE MANAGER
REVIEWED
COMMENTS AND EMAIL THREADS POSTED
CHANGE MANAGER & INITIATOR
APPROVED BY FOREST RESOURCE GROUP
PENDING APPROVAL
COMMENTS, CONCERNS, ISSUES
FRG & STEERING COMMITTEE
APPROVED BY STEERING COMMITTEE
APPROVED FOR TEST
TEST ENVIRONMENT SUCCESSFUL
YES
APPROVED FOR PRODUCTION (CLOSED)
SPECIFIC PROBLEMS OR SUCCESSES
SYS ADMINS
NO
DENIED

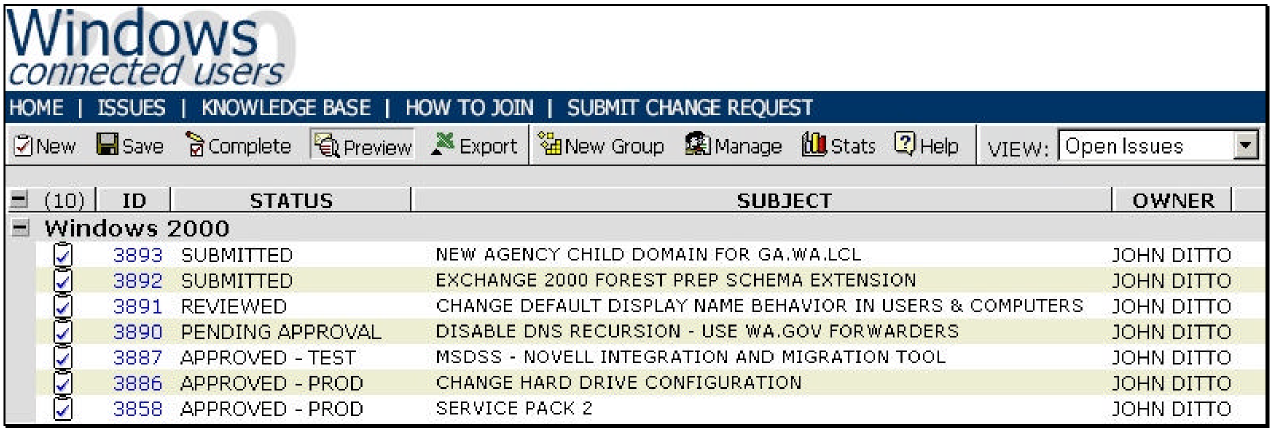
Figure 5 - Change Request Management Process

End User Design Considerations

The user experience is critical toward making the change management tool a successful part of the communications platform. Unlike the news section that will consist predominantly of readers, the most active users in the change management tool will be the owners and approvers of CRs. In addition to serving as a good communication tool, the change management tool must also provide a flexible interface for management of the requests.

This is most easily accomplished using a two-part user interface. The general section presents a list of issues based on criteria. The criteria would allow the viewer some latitude in defining his or her environment. For example, Adam may wish to see all CRs that haven't been approved for test, and are more than 30 days old. Cindy, on the other hand, might want to view all the CRs that only she has submitted.

Figure 6 illustrates what a list screen might look like. Ideally, the most critical fields (status, owner, subject and last updated) would be visible and the user would be required to click on the issue to view its details, as shown in Figure 7. The details screen would also be where change management personnel could make comments, change status, and submit approvals.



(10)	ID	STATUS	SUBJECT	OWNER
Windows 2000				
☑	3893	SUBMITTED	NEW AGENCY CHILD DOMAIN FOR GA.WA.LCL	JOHN DITTO
☑	3892	SUBMITTED	EXCHANGE 2000 FOREST PREP SCHEMA EXTENSION	JOHN DITTO
☑	3891	REVIEWED	CHANGE DEFAULT DISPLAY NAME BEHAVIOR IN USERS & COMPUTERS	JOHN DITTO
☑	3890	PENDING APPROVAL	DISABLE DNS RECURSION - USE WA.GOV FORWARDERS	JOHN DITTO
☑	3887	APPROVED - TEST	MSDSS - NOVELL INTEGRATION AND MIGRATION TOOL	JOHN DITTO
☑	3886	APPROVED - PROD	CHANGE HARD DRIVE CONFIGURATION	JOHN DITTO
☑	3858	APPROVED - PROD	SERVICE PACK 2	JOHN DITTO

Figure 6 - Change Management List Screen

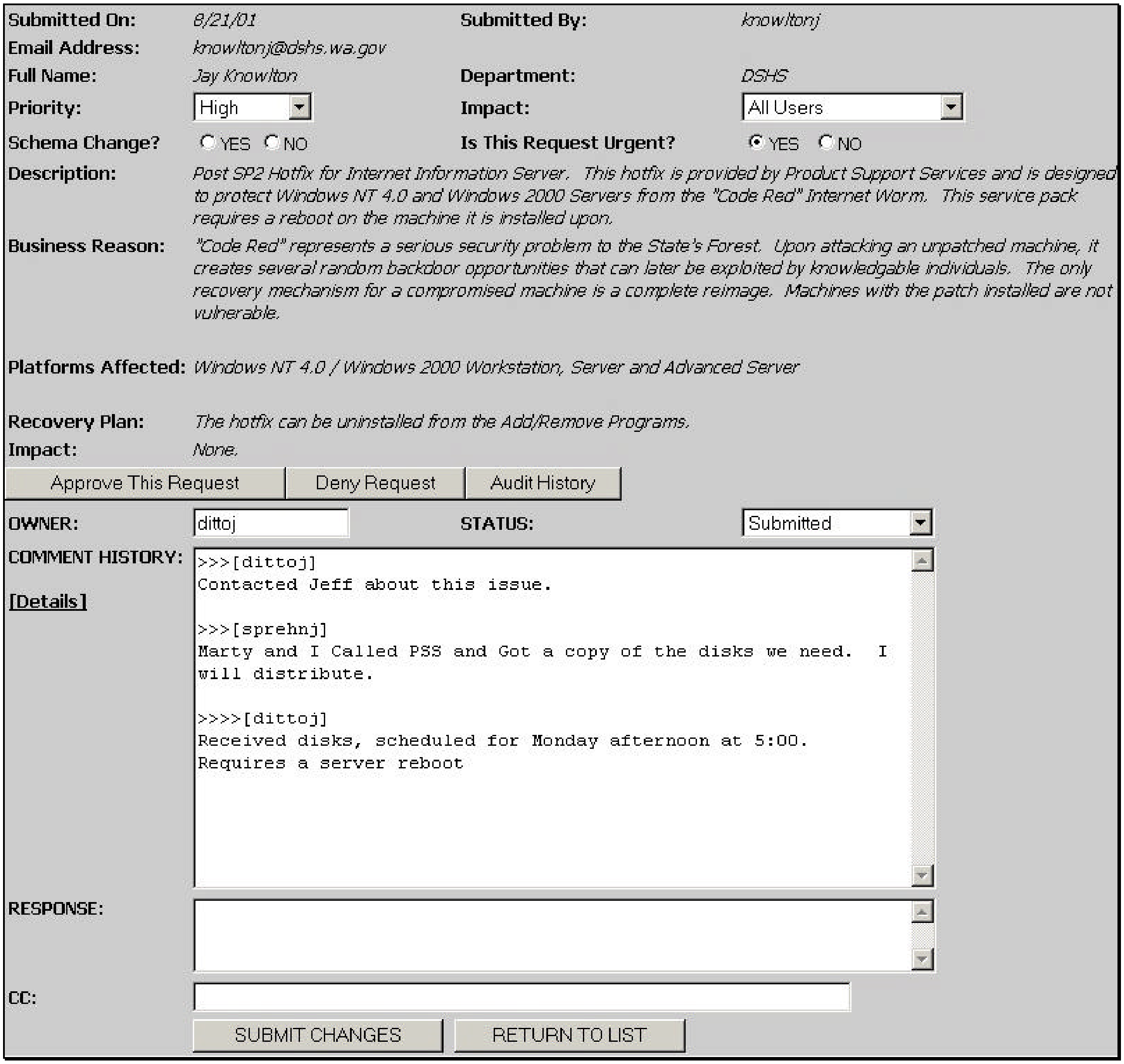
Figure 8, on the other hand, shows what the empty form might look like to a user submitting a new change request.

Standard Changes

Standard changes are defined as change management tasks that are common. Examples include the addition of a connector, adding a new agency to the forest, etc.

To facilitate standard changes, the change request tool will use special forms for each of the common tasks identified. These forms will be organized around the needs of the user, but populate the change management database on the backend, using the fields, tools, and process defined for all changes.

This gives customers a more pleasant experience, which is particularly important the first time they are working through the process. More importantly, it gives the change management team assurance that all information for a standard change has been gathered before the change was submitted. This drastically speeds up the change request process for standard changes.



Submitted On:	*8/21/01*	**Submitted By:**	*knowltonj*
Email Address:	*knowltonj@dshs.wa.gov*		
Full Name:	*Jay Knowlton*	**Department:**	*DSHS*
Priority:	High	**Impact:**	All Users
Schema Change?	○ YES ○ NO	**Is This Request Urgent?**	● YES ○ NO

Description: *Post SP2 Hotfix for Internet Information Server. This hotfix is provided by Product Support Services and is designed to protect Windows NT 4.0 and Windows 2000 Servers from the "Code Red" Internet Worm. This service pack requires a reboot on the machine it is installed upon.*

Business Reason: *"Code Red" represents a serious security problem to the State's Forest. Upon attacking an unpatched machine, it creates several random backdoor opportunities that can later be exploited by knowledgable individuals. The only recovery mechanism for a compromised machine is a complete reimage. Machines with the patch installed are not vulnerable.*

Platforms Affected: *Windows NT 4.0 / Windows 2000 Workstation, Server and Advanced Server*

Recovery Plan: *The hotfix can be uninstalled from the Add/Remove Programs.*

Impact: *None.*

OWNER: dittoj **STATUS:** Submitted

COMMENT HISTORY:

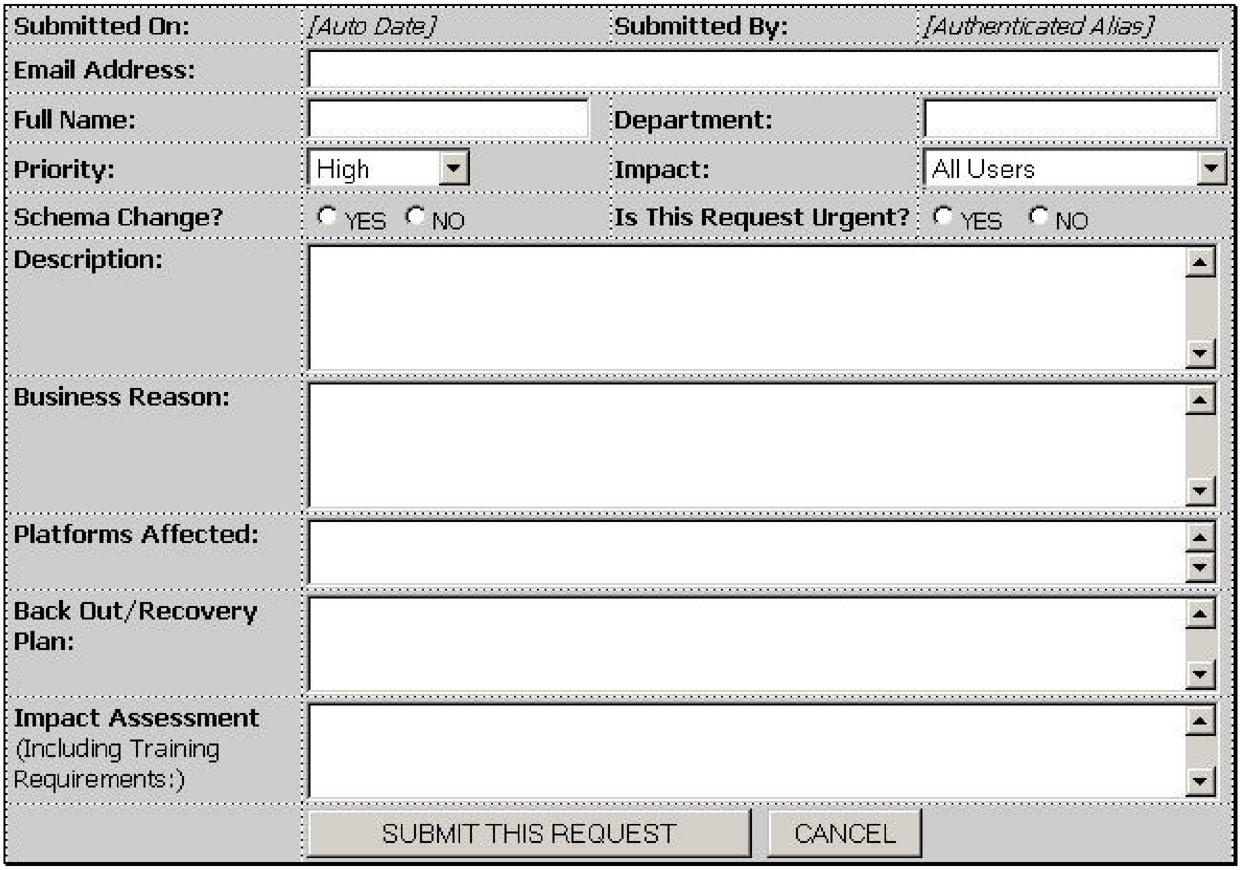
[Details]

```
>>>[dittoj]
Contacted Jeff about this issue.

>>>[sprehnj]
Marty and I Called PSS and Got a copy of the disks we need.  I
will distribute.

>>>>[dittoj]
Received disks, scheduled for Monday afternoon at 5:00.
Requires a server reboot
```

RESPONSE:

CC:

Figure 7 - Details for Change Mgt Team



Submitted On:	[Auto Date]	Submitted By:	[Authenticated Alias]
Email Address:			
Full Name:		Department:	
Priority:	High	Impact:	All Users
Schema Change?	○ YES ○ NO	Is This Request Urgent?	○ YES ○ NO
Description:			
Business Reason:			
Platforms Affected:			
Back Out/Recovery Plan:			
Impact Assessment (Including Training Requirements:)			
	SUBMIT THIS REQUEST	CANCEL	

Figure 8 - End User Change Request Form

NOT SHOWN IN Figure 8: in addition to the fields depicted above, the form will include two additional text fields that allow the change initiator to list references and dependencies. References are a comma separated list of other change request numbers that should be used by the change management team for reference purposes (the equivalent of FYI).

The dependencies text field allows an initiator to explicitly define a critical path for change requests. For example, change B is dependent on change A happening first. In the event that Change B is worked through the process, the change management team would be explicitly warned that change B depends on the completion of change A.

Published Authoritative Documents

Published documents include formal documents drafted by the Forest Resource Group and approved by the Steering Committee. Generally, they require conversion to a portable format and get published as read-only documents on the Forest Web Site.

Business Requirements

Because these documents are published rather infrequently, it has been determined by the Forest Resource Group that the current manual process is sufficient when considering the cost-benefit analysis of implementing a more automated publishing system.

The current process is documented as follows:

1. The document is assigned a version number upon initial creation that is incremented with each subsequent change.
2. Upon approval, the document is delivered by the steering committee to the project manager. The project manager is responsible for getting the document converted to a portable format (currently Adobe Acrobat).
3. Once the document has been converted and reviewed by the project manager, it is published on the web and a link to the document is added to the published document main page.

Process Workflow

Although the process for creating a new authoritative document is relatively straightforward, changes to existing documents can quickly become complicated, particularly if major revisions to sections and chapters are recommended. Figure 9 illustrates the workflow requirements identified above.

End User Design Considerations

In addition to these requirements, it is preferred that the Authoritative documents be segregated from the other content. This ensures that end users can distinguish between a best-practice and a requirement.

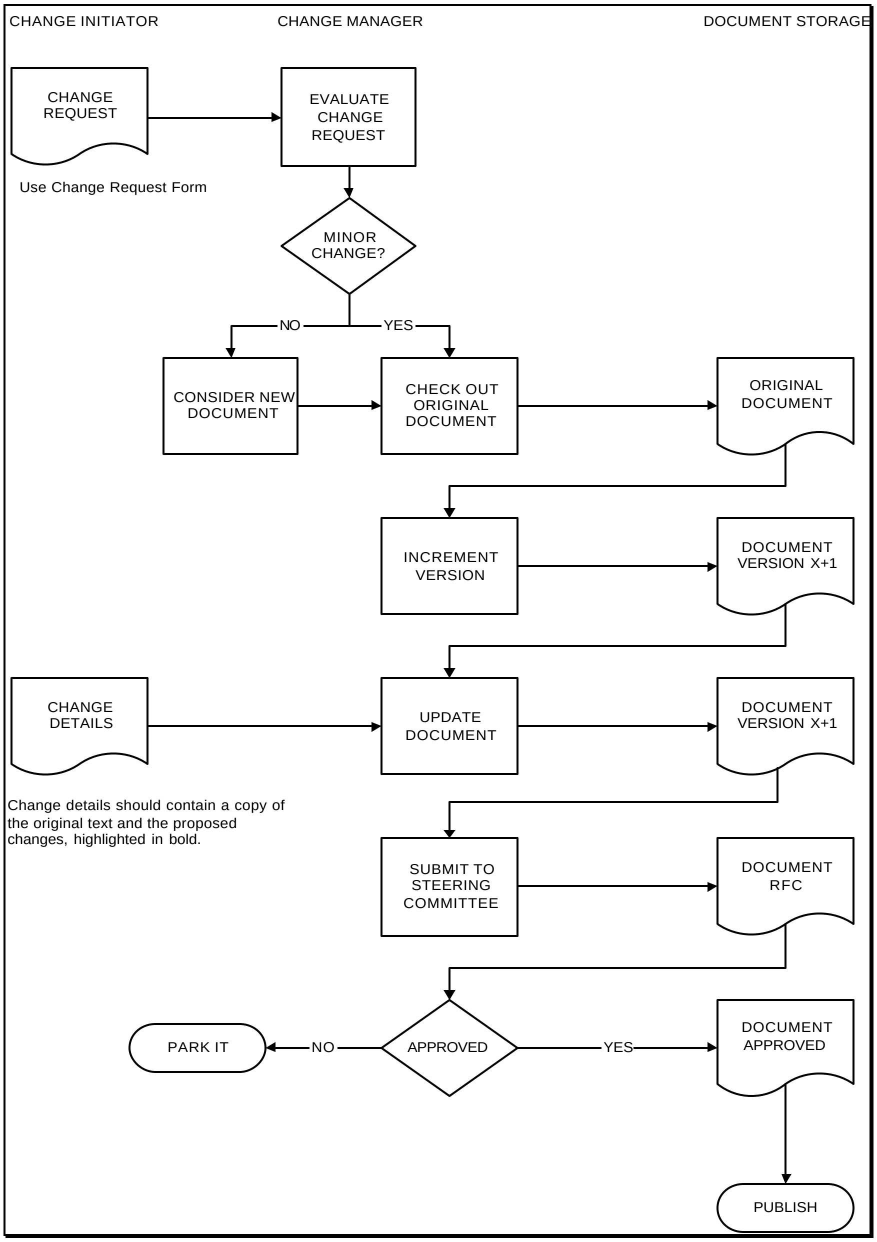


Figure 9 - Process Workflow for Authoritative Documents

Technical Knowledge Management

Technical Knowledge Management is very similar to other types of communication, with two prime differences. KM typically has a longer life span than status or communication reports and KM usually follows a structured format so that information can be more easily managed as the repository grows.

Business Requirements

The driving force behind a **knowledge management repository (KMr)** revolves around the structuring of information. Once a structure has been identified, the process and business requirements for KMr are very similar to those for published authoritative documents and news.

For this reason, the KMr section of the communications platform will focus its requirements around the document structure. The structure attempts to accomplish the following:

1. Define the elements that must exist in each document. Elements include authors, reviewers, approvers, Document Body, etc.
2. Define the attributes that can describe those elements. For example, the author element might have a name attribute and an email address attribute.

Elements

The elements listed below are named using the camelHump naming convention. This is similar to the W3C standard for XML documents. The element names are not necessarily final, but represent the identified data elements that should exist in a technical knowledge management document.

<approver>	The approver tag is an empty child element of the contributors container that identifies individuals that contribute to the approval of a document. Approvers should be listed separately from authors and contributors.
<author>	The author element is an empty child element of the contributors container that identifies individuals that contribute to the authoring of a document. Authors should be listed separately from reviewers, and approvers.
<contributors>	The contributors element is a **child** element of the docInfo container. It is used for defining the contact information for all persons involved in the authoring, reviewing and approving of the document. It actas as a wrapper element for all other elements.
<docBody>	The docBody element is a child of the document element and sibling of the docInfo element. It is analogous to the HTML tag <body>. Its purpose is to encapsulate the body of the document that would typically be exposed through publication vehicles (such as the internal web site).
<docInfo>	The docInfo element is similar to the <head> section of an

	HTML document. It is a container for document metadata. The docInfo element is a **child** of the document element.
<document>	The document element is the root element in the xml document.
<h1>|<h2>|<h3> <h4>|<h5>|<h6>	The h series of elements are analogous to the HTML header tags. They represent various headings within a document. These tags should not be used for formatting your document. They should only be used for logical structure.
<history>	The history element is a **child** element of the docInfo container. It is a container for the version element that records changes and comments through the life of the document.
<keyWord>	The keyWord element used to call attention to a word or phrase. It should be used within paragraph elements, list item elements, or table cell elements. It should not be used to wrap around multiple paragraphs or other elements.
<li>	li is a child element to the list element and analogous to the <LI> tag in HTML. Use one li element for each distinct item in a list.
<list>	The list elements are analogous to the <OL> and <UL> tags in HTML. Use it as a container for any list.
<p>	The p element is analogous to the <P> tag in HTML. Use it for identifying paragraphs of text.
<queryWords>	The queryWords element is a **child** element of the docInfo container. It is a container for the qWord elements.
<qWord>	The qWords element is a **child** element of the queryWords container. It should not be used for documenting keywords, but rather for imaginary search words. For example, “Win2K” might be used for documents about Windows 2000. “Win2K” is not a real word, but one that could be used in a search to ensure that only documents about Windows 2000 are returned.
<reference>	The reference element is analogous to the <a href> tag in HTML. It can be used for citations or hyperlinks.
<reviewer>	The reviewer element is an empty child element of the contributors container that identifies individuals that contribute to the review of a document. Reviewers should not be authors.
<section>	The section element is a child of the docBody element. It is used to break up the document into logical sections.
<table>	The table element is analogous to the <TABLE> tag in HTML. Use it to define the boundaries of a table.
<td>	The td element is analogous to the <TD> tag in HTML. Use it for identifying table cells.
<tr>	The tr element is analogous to the <TR> tag in HTML. Use it for identifying rows in a table.
<version>	The version element is an empty child element of the history container that records a specific change or comment to the document.

Attributes

Attributes are used to define data elements. For example, the author element in a document would have a name attribute and an email address attribute. Following the standard provided by Active Directory, the Technical Knowledge Management documentation will define elements and attributes independently, with the ability to link attributes to elements as needed. For each attribute, the related elements have been listed.

Limit choices refers to whether or not an author must select possible values for that attribute from a list, verses having the freedom to specify any valid text.

"Possible values" outlines the data type for the attribute. In some cases, it may be necessary to represent an attribute as a specific data type, such as number or date. In most cases, however, any valid alpha-numeric characters are considered acceptable.

agency

Description:	Use the approved agency ID documented in the Forest Naming Conventions. The size of this field is limited only by policy. Technically, the value can be of any length.
Applies To:	author, reviewer, approver elements
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

colNbr

Description:	Identifies the column in an XML-based table. Although the data-type for this attribute is any alpha-numeric value, it is recommended that letters be used to identify columns, similar to the handling of columns by Microsoft Excel.
Applies To:	td element
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

comments

Description:	Used for comments. Generally comments are not intended for a general audience, but are appropriate to specific individuals, groups or participants privy to the document.
Applies To:	all elements
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

company

Description:	Used to identify the company responsible for producing the hardware or software being documented by the article. For example, documents about Windows 2000 should have a company attribute equal to "Microsoft" in the docInfo element.
Applies To:	docInfo

Limit Choices:	no
Possible Values:	Any alpha-numeric values.

contributor

Description:	Should contain the email address of a valid contributor in the contributors section of the document.
Applies To:	version element
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

date

Description:	Defines a date and time stamp.
Applies To:	version element
Limit Choices:	Limit to valid dates with optional time
Possible Values:	yyyy/mm/dd [hh:mm:ss]

docType

Description:	Identifies the type of document that was created.
Applies To:	document
Limit Choices:	yes
Possible Values:	support | bestPractice | communication | alert

email

Description:	Contains the email address of an individual.
Applies To:	author, reviewer, approver
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

fName

Description:	The first name of an individual.
Applies To:	author, reviewer, approver
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

lName

Description:	The last name of an individual.
Applies To:	author, reviewer, approver
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

href

Description:	Must contain a valid hyperlink.
Applies To:	reference
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

product

Description:	Must contain the Official name of the product to which the document applies. For example, "Windows 2000 Server," or "Exchange Server 2000."
Applies To:	docInfo
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

rowNbr

Description:	Identifies a row in an XML-based table. Although no restrictions on usage exist, use Excel-style row numbers for consistency across all documents in the knowledge base.
Applies To:	tr
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

security

Description:	Identifies the intended audience for a document or any part within a document. If omitted, the element will be treated as public.
Applies To:	document, section, p, list, table
Limit Choices:	yes
Possible Values:	public | private | confidential

source

Description:	Used to provide the name, contact information, or bibliography information for a referenced item in the document. This should not include a URL.
Applies To:	reference
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

status

Description:	Defines the current status of a document.
Applies To:	document element
Limit Choices:	yes
Possible Values:	development | draftForComment | approved

type

Description:	Defines the type of content that appears within a designated section of a document. Note that the value "status" is different from the attribute status. The value status would be used to define the status of a particular problem, not necessarily the status of the document itself.
Applies To:	section element
Limit Choices:	yes

Possible Values:	summary | description | stepsToReproduce | symptoms | resolution | moreInformation | status

value

Description:	Used to provide an alpha-numeric value to XML elements
Applies To:	version
Limit Choices:	no
Possible Values:	Any alpha-numeric values.

version

Description:	Used to specify the version number of a particular product that is being referenced in the document. For example, if the article is related to Windows 2000, the version attribute should equal "5.0."
Applies To:	docInfo
Limit Choices:	no
Possible Values:	Any alpha-numeric values.